B.Sc. Project/Thesis Proposal

Bangladesh Army International University of Science & Technology
Cantonment, Cumilla.

Author:	**Ashaduzzaman Hasib,** **Barkat Ullah,** **Kawshar Ahmed,** **Md. Riaz Uddin Khan**	**Supervisor**	**Shahed Khan** **Lecturer, Dept. Of CSE,** **BAIUST**
E-mail:	Ashhasib54@gmail.com barkatullah382@gmail.com, riazuddin@gmail.com, kawsharahmed@gmail.com	Email:	shahed@baiust.edu.bd
Phone:	+8801762189492	Phone:	
Specialization:	Computer Vision	Defense Planned:	

Proposed Topic:

Detection of Bengali Ligatures using Deep Convolutional Neural Network

Topic Characteristics :

Bengali Ligature is formed by two or more Bengali alphabets where the newly formed compound alphabet is very different in shape and pronunciation than its origins. There are total 285 Bengali ligatures in the Bengali Language. Learners often find it difficult to process an unknown ligature while reading or writing. Moreover, there are some ligatures that may look very similar in shape. Like Bengali alphabets, the ligatures also have three types based on “Maatra”(A straight line from over the alphabet). The first one is “Purnomatra”, the 2nd one and the third one are called “Ardho Maatra” and “Maatrahin”. The proposed method uses powerful features of Artificial Intelligence to detect a written ligature or a processed image which consists of a ligature and recognize it to further show the root alphabets from where it was formed. This method of recognizing is done by a Deep Convolutional Neural Network that takes numerical presentation of pre-processed images. The main input of the Neural Network must be numerical, so the pre-processing phase uses Convolution operation on images to extract the features from it. The whole process is implemented in Google Colaboratory. The model can be found here.

Working Hypothesis:

The goal is not only to recognize a ligature letter but also to determine the root letters and the usage of it. So, overall this model is learning based.The hypothesis of the proposed model can be classified as:

1. Detection of ligatures from camera captured image: The model can be used to recognize ligatures from images via phone camera in real-time. This process can be done using any Android or iPhone Device.

2. Detection of ligatures from app: An android or iOS app having a canvas where one can write or draw on-screen can also use this Machine Learning model to recognize if it is a ligature.

In both cases, the model must be in a remote server and there must be an API connected to it in order to put image requests or get response from it.

Methodology:

All the necessary steps whether it is data pre-processing, training or testing is accomplished using Python3, more specifically Keras library which uses Tensorflow model. These types of training requires a lot of computational power, so Google Colaboratory is used for the heavy parts. The whole procedure can be broken down into –

1. Data pre-processing: To build these types of models we need training data. As a start, we are working with 64 ligatures among 285 because of lack of volunteers. The dataset consists of raw images that are actually captured by camera and converted into binary images for a better feature extraction. These converted images are rotated in 8 different angles [3,6,9,12,-3,-6,-9,-12]. Finally, the combined images are ready for the later Convolution operation.

2. Convolution: Convolution operation is used for extracting features from the data images. We don't need the whole image to recognize it, rather we use patterns to find out what is similar to the learning that is already in our minds. So, this operation extracts feature matrix that is the resultant value of the operation itself.

3. Max-Pooling: Pooling operation is applied on the convoluted images to have better extraction values. Like convolution, pooling also uses a feature map to extract features from it. The main reason of this step is to extract better feature matrix and at the same time, to reduce the size of the matrix.

4. Flattening: This step converts the pooled 2D images into a 1D array to feed the Artificial Neural Network further. So, the output of this step will be 1D array full of numerical values.

All the steps except data pre-processing are combined into a layer which is called the Fully Connected Layer before the input layer of the Actual ANN.

Here is the architecture of the ANN that is used for the model:

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 26, 26, 16)	448
batch_normalization_1 (Batch	(None, 26, 26, 16)	64
conv2d_2 (Conv2D)	(None, 24, 24, 16)	2320
batch_normalization_2 (Batch	(None, 24, 24, 16)	64
max_pooling2d_1 (MaxPooling2	(None, 12, 12, 16)	0
dropout_1 (Dropout)	(None, 12, 12, 16)	0
conv2d_3 (Conv2D)	(None, 10, 10, 32)	4640
batch_normalization_3 (Batch	(None, 10, 10, 32)	128
conv2d_4 (Conv2D)	(None, 8, 8, 32)	9248
batch_normalization_4 (Batch	(None, 8, 8, 32)	128
max_pooling2d_2 (MaxPooling2	(None, 4, 4, 32)	0
dropout_2 (Dropout)	(None, 4, 4, 32)	0
flatten_1 (Flatten)	(None, 512)	0
dense_1 (Dense)	(None, 512)	262656
dropout_3 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 1024)	525312
dropout_4 (Dropout)	(None, 1024)	0
dense_3 (Dense)	(None, 10)	10250

Total params: 815,258
Trainable params: 815,066
Non-trainable params: 192

References/Bibliography :

1) **Introduction to convolutional Neural Networks** by **Jianxin Wu** **[** LAMDA Group, National Key Lab for Novel Software Technology, Nanjing University, China (May 1, 2017) **]**

2) **Handwritten Bangla Character Recognition using Artificial Neural Network** by **Md. Azher Uddin [** IOSR journal of Computer Engineering, e-ISSN : 2278-0661, p-ISSN : 2278-8727 Volume 16, Issue 3, Ver. III(May-June, 2014) **]**

3) **Semiotic Interpretation of Bangla Ligatures : An introduction** by **Sanjid Afrin [** The Dhaka University Journal of Linguistics: Vol 2, No: 3, February 2009, Page : 111-124. ISSN: 2075-3098 **]**

4) **Recognition of Bangla Compound Characters Using Structural Decomposition** from **Researchgate.net [** Article on Pattern Recognition 47(3) : 1187-1201 **]**